IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **DeWitt, Jr. et al.**	§	
	§	Group Art Unit: **2183**
Serial No. **10/675,831**	§	
	§	Examiner: **Cody, Dillon J.**
Filed: **September 30, 2003**	§	
	§	
For: **Method and Apparatus for Generating Interrupts upon Execution of Marked Instructions and upon Access to Marked Memory Locations in a Data Processing System**	§	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

35525
PATENT TRADEMARK OFFICE
CUSTOMER NUMBER

APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on August 31, 2006.

A fee of $500.00 is required for filing an Appeal Brief. Please charge this fee to IBM Corporation Deposit Account No. 09-0447. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to IBM Corporation Deposit Account No. 09-0447. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0447.

REAL PARTY IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines Corporation of Armonk, New York.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-25

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: None
2. Claims withdrawn from consideration but not canceled: None
3. Claims pending: 1-25
4. Claims allowed: None
5. Claims rejected: 1-25
6. Claims objected to: None

C. CLAIMS ON APPEAL

The claims on appeal are: 1-25

STATUS OF AMENDMENTS

An Amendment after Final Rejection was not filed in the case. Accordingly, the claims on appeal herein are as amended in the Response to Office Action dated April 26, 2006 and as finally rejected in the Final Office Action dated June 2, 2006.

SUMMARY OF CLAIMED SUBJECT MATTER

A. CLAIM 1 - INDEPENDENT

The subject matter of claim 1 is directed to a method in a data processing system for processing instructions. Responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system (Steps **800**, **802**, **Figure 8**, and page 30, lines 5-6), it is determined whether a performance indicator that identifies that execution of the instruction is to be monitored is present (Step **804**, **Figure 8**, and page 30, lines 6-9). An interrupt is forced if the performance indicator is present (Step **806**, **Figure 8**, and page 30, lines 16-18).

B. CLAIM 8 - INDEPENDENT

The subject matter of claim 8 is directed to a method in a data processing system for processing data. Responsive to an access of data (step **900**, **Figure 9**, and page 31, lines 13-14), it is determined whether a performance indicator that identifies that access of the data is to be monitored is present (Step **902**, **Figure 9**, and page 31, lines 14-17). An interrupt is generated if the performance indicator is present (Step **904**, **Figure 9**, and page 31, lines 17-20).

C. CLAIM 14 - INDEPENDENT

The subject matter of claim 14 is directed to a data processing system for processing instructions. The system includes determining means (**214**, **Figure 2**, **300**, **Figure 3**, see also, page 15, lines 4-6 and page 23, lines 8-10) responsive to receiving an instruction (**602**, **604**, **606**, **Figure 6A**, see also page 27, line 22 – page 28, line 4) for execution in an instruction cache (**214**, **Figure 2**, **300**, **Figure 3**, see also, page 15, lines 4-6 and page 23, lines 8-10) in a processor (**210**, **Figure 2**, see also page 14, lines 17-21) in the data processing system (**100**, **Figure 1**, see also page 12, lines 2-4) for determining whether a performance indicator (**510**, **512**, **514**, **Figure 5**, see also page 27, lines 1-21) that identifies that execution of the instruction (**602**, **604**, **606**,

Figure 6A, see also page 27, line 22 – page 28, line 4) is to be monitored is present, and forcing means (**250**, **Figure 2**, see also page 21, line 28 – page 22, line 4) for forcing an interrupt if the performance indicator (**510**, **512**, **514**, **Figure 5**, see also page 27, lines 1-21) is present.

D. CLAIM 18 – INDEPENDENT

The subject matter of claim 18 is directed to a data processing system for processing data. The data processing system (**100**, **Figure 1**, see also page 12, lines 2-4) includes determining means (**210**, **Figure 2**, **400**, **Figure 4**, see also page 14, lines 17-21, and page 25, lines 5-20) responsive to an access of data (**610**, **612**, **614**, **Figure 6B**, see also page 28, lines 5-13), for determining whether a performance indicator (see page 28, lines 5-8) that identifies that access of the data (**610**, **612**, **614**, **Figure 6B**, see also page 28, lines 5-13) is to be monitored is present, and generating means (**250**, **Figure 2**, see also page 21, line 28 – page 22, line 4) for generating an interrupt if the performance indicator is present.

E. CLAIM 21 – INDEPENDENT

The subject matter of claim 21 is directed to a computer program product in a computer readable medium for processing instructions. The computer program product includes first instructions for responding to receiving an instruction for execution in an instruction cache in a processor in the data processing system (Steps **800**, **802**, **Figure 8**, page 30, lines 5-6), determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present (Step **804**, **Figure 8**, page 30, lines 6-9), and second instructions for forcing an interrupt if the performance indicator is present (Step **806**, **Figure 8**, page 30, lines 16-18).

F. CLAIM 24 – INDEPENDENT

The subject matter of claim 24 is directed to a computer program product in a computer readable medium for processing data. The computer program product includes first instructions

for responding to an access of data (step **900**, **Figure 9**, page 31, lines 13-14), determining whether a performance indicator that identifies that access of the data is to be monitored is present (Step **902**, **Figure 9**, page 31, lines 14-17), and second instructions for generating an interrupt if the performance indicator is present (Step **904**, **Figure 9**, page 31, lines 17-20).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. GROUND OF REJECTION 1 (Claims 1-5 and 8-25)

Claims 1-5 and 8-25 stand rejected under 35 U.S.C. § 102(b) as being anticipated by Heisch (U.S. Patent No. 5,774,724).

B. GROUND OF REJECTION 2 (Claims 1, 6 and 7)

Claims 1, 6 and 7 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Short, K.L, "*Embedded Microprocessor Systems Design: An Introduction Using the Intel 80C188EB*", Prentiss-Hall, Inc., 1998, page 761 (hereinafter "Short") in view of Heisch (U.S. Patent No. 5,774,724).

C. GROUND OF REJECTION 3 (Claims 21-25)

Claims 21-25 stand rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter.

ARGUMENT

A. GROUND OF REJECTION 1 (Claims 1-5 and 8-25)

Claims 1-5 and 8-25 stand rejected under 35 U.S.C. § 102(b) as being anticipated by Heisch (U.S. Patent No. 5,774,724).

A.1 Claims 1-5, 14-17 and 21-23

In finally rejecting the claims, the Examiner states:

> As per claim 1, Heisch discloses a method in a data processing system for processing instructions, the method comprising: responsive to receiving an instruction for execution in an instruction cache (Fig. 2 cache 60) in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present (Col. 5 lines 58-65); and forcing an interrupt if the performance indicator is present (Col. 5 line 66-col. 6 line 6). *The examiner asserts that matching the contents of the IABR register to the executing instruction's address constitutes a "performance indicator". Col. 4 lines 13-16 disclose that upon matching a start address to the IABR, performance monitoring is initiated by means of an exception.*

Final Office action dated June 2, 2006, page 3.

Claim 1 on appeal herein is as follows:

> 1. A method in a data processing system for processing instructions, the method comprising:
>
> responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present; and
>
> forcing an interrupt if the performance indicator is present.

A prior art reference anticipates a claimed invention under 35 U.S.C. § 102 only if every element of the claimed invention is identically shown in that single prior art reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of a claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983).

Appellants respectfully submit that Heisch does not identically show every element of the claimed invention arranged as they are in the claims; and, accordingly, does not anticipate the claims. With respect to claim 1, in particular, Heisch does not teach or suggest "responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present".

Heisch is directed to performance monitoring of computer systems. In particular, Heisch describes a mechanism by which a performance monitor and an instruction address breakpoint facility are combined to provide finer granularity in performance monitoring. As described in col. 3, line 67 – col. 4, line 7 of Heisch:

> An instruction address breakpoint may be set for any effective address in a program utilizing an instruction address breakpoint register (IABR). The IABR will cause an instruction address breakpoint exception when the instruction at that address is executed. By coupling the performance monitor and IABR functionality, the performance monitor may thereby be enabled and disabled on a per instruction address basis.

The Examiner refers to col. 5, lines 58-65 of Heisch as disclosing "responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present". Appellants respectfully disagree. Col. 5, lines 58-65 of Heisch is as follows:

> The basic operation of an IAB is as follows. First, an address of interest may be loaded into the IAB register, whereupon the microprocessor associated with the IABR executes program instructions while the register is monitored. The contents of the IAB register, more specifically, are compared to the address of the particular instruction executing at a given time. This function is shown by the IABR compare block **64** of FIG. 2.

Nowhere in the above recitation, nor anywhere else in Heisch, is there any disclosure or suggestion of making a determination as to whether a performance indicator that identifies that execution of an instruction is to be monitored is present in response to receiving an instruction for execution in an instruction cache in a processor. The above recitation in Heisch states only that contents of the IAB register are compared to an address of a particular instruction that is

executing. Heisch does not determine whether a performance indicator is present that identifies that execution of an instruction is to be monitored in response to receiving the instruction for execution.

In col. 8, lines 16-24, Heisch states as follows:

> The purpose of the lines such as **66**, **68**, **70**, **71** is to indicate that, as previously described, the performance monitor **42** may be programmed as desired to generate counts associated with a number of respective parameters, the particular selection of these parameters being pre-programmed. Thus, if it is desired to count cache misses, this interconnection between the performance monitor **42** and cache **60** illustrated by arrow **71** indicates that such a statistic may be generated by the performance monitor **42** as desired. (Emphasis added.)

This paragraph states that cache misses may be counted by the performance monitor "if it is desired to count cache misses". This statement suggests that Heisch does not determine whether a performance indicator that identifies that execution of an instruction is to be monitored is present before forcing an interrupt and supports Appellants' assertion that Heisch does not disclose or suggest "responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present".

Heisch, accordingly, does not show every element of the claimed invention arranged as they are in claim 1, and does not anticipate claim 1.

Furthermore, in rejecting the claims, the Examiner asserts that matching the contents of the IABR to the executing instruction's address constitutes a "performance indicator". Specifically, in Heisch, the IABR is monitored to determine if a preselected address stored in the IABR equals an address of an instruction currently to be executed. An interrupt is caused if there is such a match. If the match itself is construed as comprising the performance indicator of claim 1, as proposed by the Examiner, such a match does not identify that execution of the instruction is to be monitored is present as recited in claim 1. In Heisch, all instructions are monitored and any "indicator" that may be present in Heisch may indicate whether there is to be an interrupt, but does not indicate that execution of an instruction is to be monitored is present. For this reason as well, Heisch does not disclose or suggest "responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a

performance indicator that identifies that execution of the instruction is to be monitored is present".

For at least all the above reasons, claim 1 is not anticipated by Heisch, and patentably distinguishes over Heisch in its present form.

Claims 2-5 depend from and further restrict claim 1 and are also not anticipated by Heisch, at least by virtue of their dependency.

Independent claims 14 and 21 recite similar subject matter as claim 1, and are also not anticipated by Heisch for similar reasons as discussed above with respect to claim 1. Claims 15-17 depend from and further restrict claim 14, and claims 22 and 23 depend from and further restrict claim 21 and are also not anticipated by Heisch, at least by virtue of their dependency.

A.2 Claims 8-13, 18-20 and 24-25

Independent claim 8 is as follows:

> 8. A method in a data processing system for processing data, the method comprising:
> responsive to an access of data, determining whether a performance indicator that identifies that access of the data is to be monitored is present; and
> generating an interrupt if the performance indicator is present.

In rejecting claim 8, the Examiner states, similar to claim 1:

> As per claim 8, Heisch discloses a method in a data processing system for processing data, the method comprising: responsive to an access of data, determining whether a performance indicator that identifies that access of the data is to be monitored is present (Col. 5 lines 58-65); and generating an interrupt if the performance indicator is present (Col. 5 line 66-col. 6 line 6). *The Examiner asserts that the method described by Heisch anticipates causing an interrupt on a data access (Col. 10 line 13-15). The examiner asserts that matching the contents of the IABR register to the address in question constitutes a "performance indicator". Col. 4 lines 13-16 disclose that upon matching a start address to the IABR, performance monitoring is initiated by means of an exception.*

Final Office Action dated June 2, 2006, pages 4-5.

As discussed in detail above with respect to claim 1, Heisch employs an IABR that stores preselected instruction address breakpoints and causes an interrupt whenever an address of an instruction currently to be executed matches a stored address. Heisch

does not disclose or suggest "responsive to an access of data, determining whether a performance indicator that identifies that access of the data is to be monitored is present". Heisch never makes such a determination, and claim 8 is also not anticipated by Heisch.

Further, even if matching of the contents of the IABR to the address in question can be construed as comprising a performance indicator as suggested by the Examiner, Heisch still does not disclose "responsive to an access of data, determining whether a performance indicator that identifies that access of the data is to be monitored is present" as recited in claim 8. Again, in Heisch, all instructions are monitored, and any such indicator would not identify "that access of the data is to be monitored is present".

For at least all the above reasons, claim 8 is also not anticipated by Heisch and patentably distinguishes over Heisch in its present form.

Claims 9-13 depend from and further restrict claim 8 and are also not anticipated by Heisch, at least by virtue of their dependency. Independent claims 18 and 24 recite similar subject matter as claim 8, and are also not anticipated by Heisch for similar reasons as discussed above with respect to claim 8. Claims 19 and 20 depend from and further restrict claim 18, and claim 25 depends from and further restricts claim 24. These claims are also not anticipated by Heisch, at least by virtue of their dependency.

Therefore, claims 1-5 and 8-25 are not anticipated by Heisch, and it is respectfully requested that the Board reverse the Examiner's Final Rejection of those claims.

B. GROUND OF REJECTION 2 (Claims 1 and 6-7)

Claims 1, 6 and 7 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Short, K.L, "*Embedded Microprocessor Systems Design: An Introduction Using the Intel 80C188EB*", Prentiss-Hall, Inc., 1998, page 761 (hereinafter "Short") in view of Heisch.

In rejecting the claims, the Examiner states:

> 31. As per claim 1, Short discloses a method in a data processing system for processing instructions, the method comprising: responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether an indicator is present; and forcing an interrupt if the indicator is present. *The examiner asserts that if the opcode of the instruction (indicator) indicates the Interrupt instruction (Short pg. 761), an interrupt will be forced.*

32. Short fails to disclose wherein the indicator is a performance indicator that identifies that execution of the instruction is to be monitored.
33. Heisch discloses using an interrupt routine to monitor performance of an instruction in a microprocessor system (Col. 4, lines 8-18)
34. Heisch teaches that using an interrupt (exception) routine to handle performance monitoring gives greater flexibility and more detailed information than existing performance monitor systems (Col. 1-Col. 4)
35. It would have been obvious to one of ordinary skill in the art at the time of invention to have included Heisch's performance monitoring interrupt handler in Short's processor for the benefit of a detailed and flexible performance monitoring scheme by means of interrupts.

Final Office Action dated June 2, 2006, page 9.

Appellants respectfully submit that neither Short nor Heisch nor their combination discloses or suggests "responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present", and that claim 1 is not obvious over Short in view of Heisch.

The failure of Heisch to disclose or suggest "responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present" has been discussed in detail above. Notwithstanding the Examiner's assertion to the contrary, Short also does not disclose or suggest such a step. Short appears to briefly describe an "interrupt-type" procedure. The Examiner asserts that Short discloses that if the opcode of the instruction (indicator) indicates the interrupt instruction, an interrupt will be forced. However, an opcode indicating an interrupt instruction is not the same as and is in no way a disclosure of "determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present" responsive to "receiving an instruction for execution in an instruction cache in a processor in the data processing system", nor has the Examiner specifically indicated where this subject matter is disclosed in Short. Only the present application contains such a disclosure, and the Examiner appears to be using hindsight based on Appellants' own disclosure, and not any teachings in the references themselves, in rejecting the claims.

Furthermore, the Examiner acknowledges that Short fails to disclose "wherein the indicator is a performance indicator that identifies that execution of the instruction is to be

monitored" and cites Heisch as supplying this deficiency. As pointed out above, however, in Heisch, all instructions are monitored and any "indicator" that may be present in Heisch may indicate whether there is to be an interrupt, but does not indicate that execution of an instruction is to be monitored is present. Therefore, claim 1 is not obvious over Short in view of Heisch for this reason as well.

Claims 6 and 7 depend from and further restrict claim 1, and are also not obvious over Short in view of Heisch, at least by virtue of their dependency.

Therefore, Appellants respectfully submit that claims 1, 6 and 7 patentably distinguish over Short in view of Heisch, and it is respectfully requested that the Board reverse the Examiner's Final Rejection of those claims.

C. GROUND OF REJECTION 3 (Claims 21-25)

Claims 21-25 stand rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. In rejecting the claims, the Examiner states:

> Pages 64-65 of the specification define "computer readable media" to include "transmission-type media". Transmission media are not tangible, and hence, non-statutory. *The examiner notes that the amendment to the specification filed 1 May 2006 has NOT been entered, as it constitutes new matter. In order to overcome this rejection, the examiner recommends amending claims 19-25 to read "A computer program product in a computer readable recordable-type medium..."*

Final Office Action dated June 2, 2006, page 11.

This rejection is respectfully traversed. The Examiner asserts that claims 21-25 are not limited to tangible embodiments. No basis is present, however, for holding a computer usable medium claim non-statutory because the medium may be allegedly "intangible." The MPEP states:

> In this context, "functional descriptive material" consists of **data structures** and computer programs **which impart functionality when employed as a computer component**. (The definition of "data structure" is "a physical or logical relationship among data elements, designed to support specific data manipulation functions." The New IEEE Standard Dictionary of Electrical and Electronics Terms 308 (5th ed. 1993).) "Nonfunctional descriptive material" includes but is not limited to music, literary works and a compilation or mere arrangement of data.

> **When functional descriptive material is recorded on some computer-readable medium it becomes structurally and functionally interrelated to the medium and will be statutory in most cases since use of technology permits the function of the descriptive material to be realized.** Compare *In re Lowry*, 32 F.3d 1579, 1583-84, 32 USPQ2d 1031, 1035 (Fed. Cir. 1994) (claim to data structure stored on a computer readable medium that increases computer efficiency held statutory) and *Warmerdam*, 33 F.3d at 1360-61, 31 USPQ2d at 1759 (claim to computer having a specific data structure stored in memory held statutory product-by-process claim) with *Warmerdam*, 33 F.3d at 1361, 31 USPQ2d at 1760 (claim to a data structure *per se* held nonstatutory). **(emphasis added)**

MPEP 2106 (IV)(B)(1).

Claims 21-25 recite clearly functional descriptive material since they impart functionality when employed as a computer component. Moreover, the functional descriptive material of claims 21-25 is recorded on "some" computer-readable medium.

In the above context, the term "some" means "any" computer-readable medium. The MPEP does not draw any distinctions between one type of media that is considered to be statutory and another type of media that is considered to be non-statutory. To the contrary, the MPEP clearly states that as long as the functional descriptive material is in "some" computer-readable medium, it should be considered statutory. The only exceptions to this statement in the MPEP are functional descriptive material that does not generate a useful, concrete and tangible result, e.g., functional descriptive material composed completely of pure mathematical concepts that provide no practical result. Claims 21-25 clearly recite a useful, concrete and tangible result in that an interrupt is forced or generated if a performance indicator is present. This is not just some disembodied mathematical concept or abstract idea.

Thus, claims 21-25 are directed to functional descriptive material that provides a useful, concrete and tangible result, and which is embodied on "some" computer-readable medium. Therefore, claims 21-25 are statutory and fully satisfy the requirements of 35 U.S.C. § 101. It is, accordingly, respectfully requested that the Board reverse the Examiner's Final Rejection of those claims.

/Gerald H. Glanzman/
Gerald H. Glanzman
Reg. No. 25,035
YEE & ASSOCIATES, P.C.
PO Box 802333
Dallas, TX 75380
(972) 385-8777

CLAIMS APPENDIX

The text of the claims involved in the appeal are:

1. A method in a data processing system for processing instructions, the method comprising:

responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present; and

forcing an interrupt if the performance indicator is present.

2. The method of claim 1, wherein the forcing step comprises:

sending a signal from an instruction cache to an interrupt unit in the processor; and

processing the interrupt in the interrupt unit in response to receiving the signal at the interrupt unit.

3. The method of claim 2, wherein the processing step includes:

executing code associated with the interrupt.

4. The method of claim 3, wherein the code records cache misses by a functional unit attempting to access instructions in a cache.

5. The method of claim 1, wherein the performance indicator is located in a shadow memory.

6. The method of claim 1, wherein the instruction is received in a bundle and wherein the performance indicator comprises at least one bit in a field in the bundle.

7. The method of claim 1, wherein the performance indicator is located in a field in the instruction.

8. A method in a data processing system for processing data, the method comprising:

responsive to an access of data, determining whether a performance indicator that identifies that access of the data is to be monitored is present; and

generating an interrupt if the performance indicator is present.

9. The method of claim 8, wherein the generating step comprises:

generating a signal by a data cache in which the data is located; and

receiving the signal generated by the data cache at an interrupt unit, wherein the signal indicates a presence of the interrupt to the interrupt unit.

10. The method of claim 8 further comprising:

processing the interrupt in an interrupt unit in response to generation of the interrupt.

11. The method of claim 10, wherein the processing step comprises:

executing a code for handling the interrupt.

12. The method of claim 8, wherein the performance indicator identifies that access of the data is to be monitored through a specific value in a memory location for the data.

13. The method of claim 8, wherein the data is located in a memory location.

14. A data processing system for processing instructions, the data processing system comprising:

determining means, responsive to receiving an instruction for execution in an instruction cache in a processor in the data processing system, for determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present; and

forcing means for forcing an interrupt if the performance indicator is present.

15. The data processing system of claim 14, wherein the forcing means comprises:

sending means for sending a signal from an instruction cache to an interrupt unit in the processor; and

processing means for processing the interrupt in the interrupt unit in response to receiving the signal at the interrupt unit.

16. The data processing system of claim 15, wherein the processing means includes:

executing means for executing code associated with the interrupt.

17. The data processing system of claim 16, wherein the code records cache misses by a functional unit attempting to access instructions in a cache.

18. A data processing system for processing data, the data processing system comprising:

determining means, responsive to an access of data, for determining whether a performance indicator that identifies that access of the data is to be monitored is present; and

generating means for generating an interrupt if the performance indicator is present.

19. The data processing system of claim 18, wherein the generating means comprises:

generating means for generating a signal by a data cache in which the data is located; and

receiving means for receiving the signal generated by the data cache at an interrupt unit, wherein the signal indicates a presence of the interrupt to the interrupt unit.

20. The data processing system of claim 18 further comprising:

processing means for processing the interrupt in an interrupt unit in response to generation of the interrupt.

21. A computer program product in a computer readable medium for processing instructions, the computer program product comprising:

first instructions for responding to receiving an instruction for execution in an instruction cache in a processor in the data processing system, determining whether a performance indicator that identifies that execution of the instruction is to be monitored is present; and

second instructions for forcing an interrupt if the performance indicator is present.

22. The computer program product of claim 21, wherein the second instructions include:

third instructions for sending a signal from an instruction cache to an interrupt unit in the processor; and

fourth instructions for processing the interrupt in the interrupt unit in response to receiving the signal at the interrupt unit.

23. The computer program product of claim 22, wherein the fourth instructions include;

sub-instructions for executing code associated with the interrupt.

24. A computer program product in a computer readable medium for processing data, the computer program product comprising:

first instructions for responding to an access of data, determining whether a performance indicator that identifies that access of the data is to be monitored is present; and

second instructions for generating an interrupt if the performance indicator is present.

25. The computer program product of claim 24, wherein the second instructions comprise:

first sub-instructions for generating a signal by a data cache in which the data is located; and

second sub-instructions for receiving the signal generated by the data cache at an interrupt unit, wherein the signal indicates a presence of the interrupt to the interrupt unit.

EVIDENCE APPENDIX

There is no evidence to be presented.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings.